VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ
FACULTY OF MECHANICAL ENGINEERING

ÚSTAV MATEMATIKY
INSTITUTE OF MATHEMATICS

OPTIMALIZACE DOPRAVNĚ-INŽENÝRSKÝCH ÚLOH
OPTIMIZATION OF TRAFFIC ENGINEERING PROBLEMS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

HUY LE

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. DUŠAN HRABEC, Ph.D.

BRNO 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ FAKULTA STROJNÍHO INŽENÝRSTVÍ

Zadání bakalářské práce

Ústav:	Ústav matematiky
Student:	**Huy Le**
Studijní program:	Aplikované vědy v inženýrství
Studijní obor:	Matematické inženýrství
Vedoucí práce:	**Ing. Dušan Hrabec, Ph.D.**
Akademický rok:	2017/18

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Optimalizace dopravně-inženýrských úloh

Stručná charakteristika problematiky úkolu:

Student si osvojí znalosti matematické optimalizace pro vybranou oblast dopravně-inženýrských úloh (např. optimalizace řízení křižovatek). Vybrané úlohy matematicky namodeluje a s využitím matematického programování, zvoleného (případně simulačního) softwaru a algoritmizace úlohy analyzuje (i z teoretického hlediska) a vyřeší.

Cíle bakalářské práce:

1. Student si zvolí aplikační inženýrskou oblast a teoreticky zpracuje úvodní část práce. Tato část bude obsahovat i (alespoň stručnou) literární rešerši.
2. Následně namodeluje vybranou/é optimalizační úlohu/y.
3. Tyto úlohy zanalyzuje a vyřeší za pomocí zvoleného optimalizačního přístupu.
4. Výsledky budou diskutovány a student navrhne další možná rozšíření pro (potenciální) budoucí práci.

Seznam doporučené literatury:

KLAPKA, J. a kol. Metody operačního výzkumu. Brno: VUT v Brně, 2001.

SKLENÁŘ, J. Simulation. Malta: University Of Malta, 2002.

GUBERINIC, S., G. SENBORN a A. LAZIC. Optimal Traffic Control: Urban Intersections. CRC Press, 2008.

BERÁNEK, Z. Heuristické optimalizační algoritmy v simulačních modelech. Brno: VUT v Brně, 2007, Diplomová práce.

HÁJKOVÁ, B. Dopravně inženýrské řešení křižovatky ulic Přímá - Nábřeží ve Zlíně.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2017/18.

V Brně, dne 27. 10. 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA STROJNÍHO INŽENÝRSTVÍ
DĚKANÁT
11

L. S.

prof. RNDr. Josef Šlapal, CSc.
ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
děkan fakulty

Abstrakt
Tato bakalářská práce se zabývá optimalizací řízení světelných signalizačních zařízení křižovatky, přesněji hledá optimální minimální signální plán příslušné křižovatky za splnění daných předpokladů a požadavků. Teoretická část se zaměřuje na uvedení předpokladů a základních pojmů, dále také podrobněji charakterizuje úlohu a popisuje použitelné algoritmy na tuto úlohu. Praktická část se věnuje aplikací těchto algoritmů na specifickém příkladu křižovatky a následně získané výsledky srovnává.

Summary
This bachelor thesis focuses on optimization of traffic light control of intersection, more specifically looks for optimal minimal signal plan of intersection considering given requirements. Theoretical part is about suppositions and basic definitions, after that usable algorithms for this problem are described in details. Practical part deals with application of these algorithms on specific example of intersection and compares obtained results afterwards.

Klíčová slova
Optimalizace, signální plán, fázové schéma, fázový přechod, tabulka mezičasů, barvení grafu, klika

Keywords
Optimization, signal plan, stage diagram, stage transition, conflict matrix, graph coloring, clique

LE, H. *Optimalizace dopravně-inženýrských úloh.* Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2018. 38 s. Vedoucí bakalářské práce Ing. Dušan Hrabec, Ph.D.

Prohlašuji, že jsem tuto bakalářskou práci zpracoval samostatně podle pokynů vedoucího bakalářské práce a s použitím uvedené literatury.

Huy Le

Děkuji svému vedoucímu práce Ing. Dušanovi Hrabci, Ph.D. za vedení mé bakalářské práce.

Huy Le

Obsah

1 Úvod

Město je geograficky vymezený útvar, na kterém se oproti venkovním oblastem státu koncentruje velká část obyvatelstva. V dnešní době téměř každá rodina vlastní alespoň jeden automobil, neboť dopravit se z jednoho konce města na druhý konec města je tímto způsobem většinou nejrychlejší. S narůstajícím počtem automobilů ale narůstá i počet automobilových nehod a stav ovzduší se kvůli emisím stále zhoršuje. Tyto nechtěné důsledky se snažíme co nejvíce redukovat a světelně řízené křižovatky jsou jedním z více řešení. Jelikož mnoho měst slouží jako důležitý dopravní uzel, je v současnosti pro město nutností mít v infrastruktuře zakomponované světelně řízené křižovatky. V opačném případě hrozí, že se města „ucpou“ a to znamená další problémy.

To, že se některé křižovatky řídí pomocí světelně signalizačních zařízení, ještě neznamená, že doprava ve městech plyne dokonale hladce a bez problémů. Nejenom velkoměsta se totiž potýkají s houstnoucí dopravou, se kterou se nemuselo při počátečním plánování křižovatky počítat. Doprava tak často kolabuje a to v různých denních dobách, což opět není žádoucí. Proto je optimalizace světelných signalizačních zařízení křižovatky velmi důležitým a také levným, poněvadž nejsou potřeba žádné velké přestavby křižovatky, řešením.

Úplně první semafor se objevil v Londýně před budovou britského parlamentu, byl instalován roku 1868, viz [1]. Sloužil k usměrnění koňských povozů a chodců. Tento semafor ještě nebyl světelný, nýbrž mechanický. První elektricky a světelně řízený semafor se objevil v roce 1914 v Clevelandu v Americe. Semafor měl zelená a červená světla, vydával zvukové signály a vyžadoval obsluhující personál. V Evropě byl první světelný semafor uveden do provozu v Paříži v roce 1922. První československý semafor byl spuštěn roku 1927 a stál na křižovatce Hybernská, Dlážděná a Havlíčkova u dnešního Masarykova nádraží v Praze. Na československých silnicích se semafory v takové podobě, jak je známe dnes, začaly objevovat až v roce 1967.

Každá křižovatka má svůj tzv. signální plán, který určuje pořadí a délku signálů volno všem dopravním proudům křižovatky. Cílem této bakalářské práce je optimalizovat fázové schéma signálního plánu na křižovatce. Výstupem bude vytvořený minimální signální plán dané křižovatky, tedy časové intervaly signálů jednotlivých proudů. Vyberou se proto vhodné metody aplikovatelné na tuto úlohu. Celý postup lze rozdělit do dvou částí. V prvním kroku se práce zaměří na optimální vygenerování fází a určení jejich optimálního počtu. V druhém kroku se bude hledat optimální pořadí vygenerovaných fází.

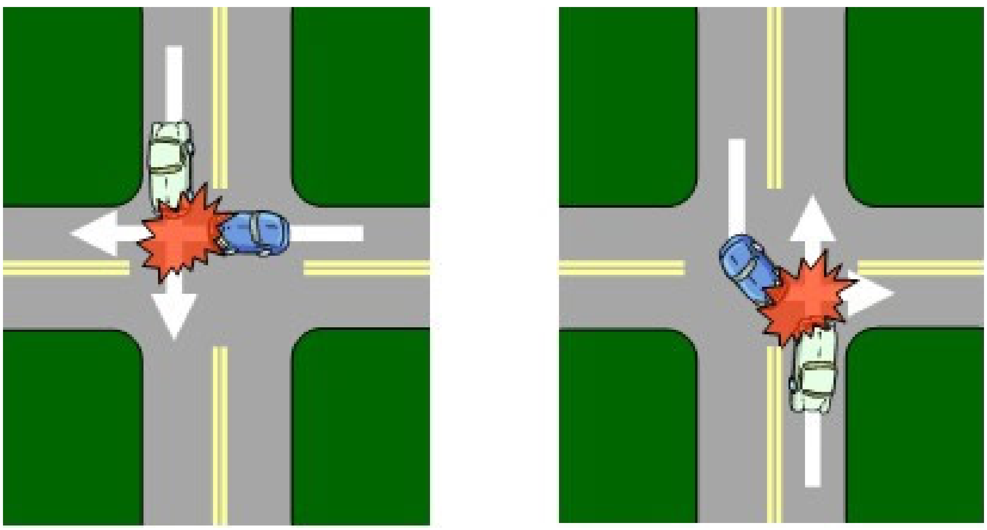
Tuto tématiku jsem si vybral, poněvadž mi obecně optimalizace přijde jako široce využitelná a prospěšná. Dále také proto, že je její aplikace lehce představitelná. V problematice světelně řízených křižovatek zde vidím prostor pro zefektivnění, jelikož stále nastávají situace, kdy se vybere jakékoliv řešení splňující požadavky, ale není přitom optimálním řešením. Dochází tak ke zbytečným ekonomickým a časovým ztrátám jak státu (při špatném nastavení signálního plánu totiž může snadno docházet k přetížení křižovatky, dopravu je pak nutno řídit ručně), tak účastníků křižovatky.

2 Pojmy a charakteristika úlohy

2.1 Základní pojmy z oblasti dopravy

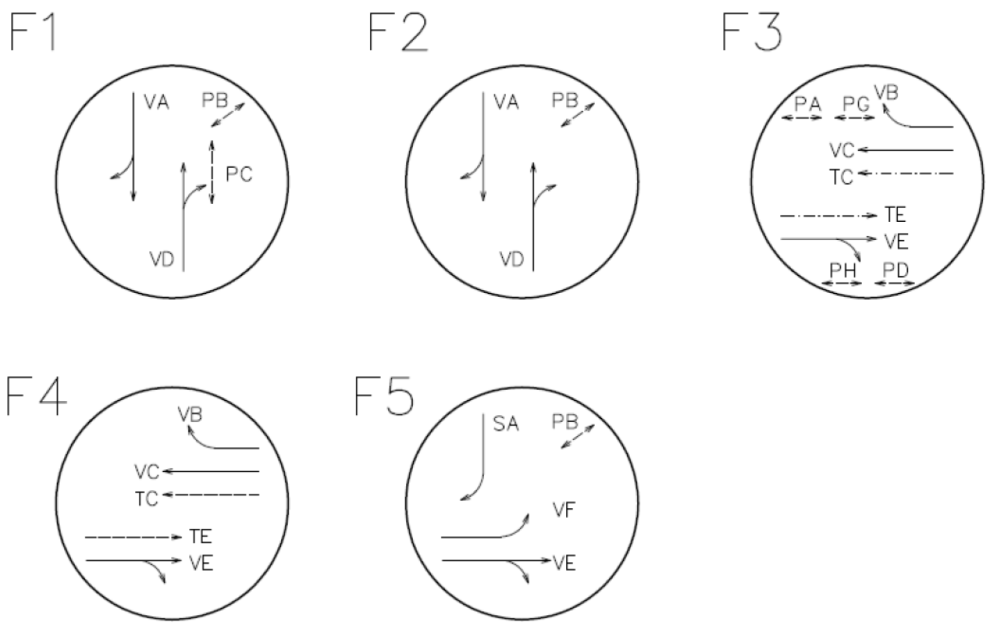
V této podkapitole budou uvedeny důležité pojmy z dopravního inženýrství, které se v práci vyskytují nebo je podstatné je pochopit. Následující definice jsou čerpány z [2].

- *Signál Stůj* je signál s červeným světlem a znamená povinnost účastníka křižovatky zastavit.
- *Signál Pozor* má dvě varianty. *Signál se současně svítícím červeným a žlutým světlem* znamená povinnost připravit se k jízdě. *Signál se žlutým světlem* znamená povinnost zastavit vozidlo. Je-li však vozidlo moc blízko, smí řidič pokračovat v jízdě. Toto se zohledňuje v signálním plánu křižovatky.
- *Signál Volno* má také dvě varianty. *Signál s plným světlem* znamená možnost pokračovat v jízdě/chůzi. *Signál se směrovou šipkou* znamená možnost pokračovat v jízdě jen ve směru šipky.
- *Signální skupina* je soubor návěstidel, která udávají v každém okamžiku pro jeden vjezd vozidel nebo vstup chodců na jeden přechod stejný signální obraz. Signální skupinu může tvořit i jediné návěstidlo.
- *Pevné řízení provozu* - jeho principem je neměnné řízení v daném časovém období. Předpokladem je ustálená intenzita provozu vozidel s pouze dlouhodobými změnami, neboť při tomto způsobu řízení není reagováno na krátkodobé výkyvy intenzity provozu. Provozně technické náklady k uskutečnění nejsou vysoké.
- *Izolovaná křižovatka* je z pohledu řízení nezávislá na vedlejších křižovatkách.
- *Dopravní proud* je pohyb vozidel, respektive chodců, za sebou nebo v pruzích vedle sebe v jednom směru (může se skládat i z několika jízdních, resp. pěších, pruhů).
- *Kolizní dopravní pohyby* jsou vzájemné pohyby vozidel (nebo vozidel a chodců), které se kříží nebo připojují (ne ty, které se odpojují). Viz obrázek 2.1.
- *Podmíněně kolizní dopravní pohyby* jsou pohyby, pro něž platí pravidla o přednosti v jízdě podle právního předpisu, např. odbočení vlevo. Viz obrázek 2.1.



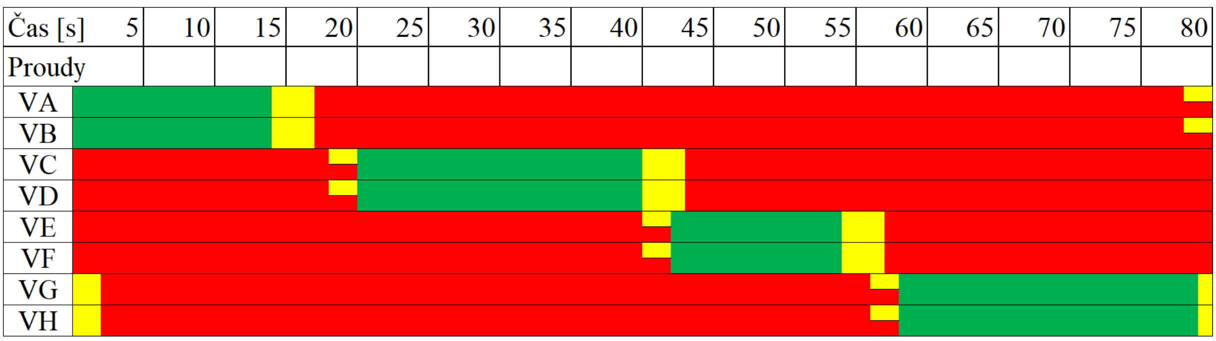
Obrázek 2.1: Vztah kolize (vlevo) a podmíněné kolize (vpravo)
Zdroj: safety.fhwa.dot.gov

- *Nekolizní dopravní pohyby* jsou všechny ostatní dopravní pohyby.
- *Kolizní plocha* je ta část plochy komunikace, kde se dráha vyklizujícího vozidla nebo chodce střetává s dráhou najíždějícího vozidla nebo chodce.
- *Signální fáze* je časový interval, ve kterém mají současně volno určité, zpravidla nekolizní dopravní pohyby na křižovatce.
- *Fázový přechod* je přechod z jedné fáze do druhé, tedy časový úsek mezi signály volno skupin končící fáze a signály volno skupin nastupující fáze. Obsahuje alespoň mezičasy kolizních směrů.
- *Fázové schéma* je přiřazení dopravních pohybů jednotlivým fázím a nejvýhodnější pořadí fází. Viz obrázek 2.2.



Obrázek 2.2: Fázové schéma
Zdroj: [2]

- *Signální plán* je program řízení světelného signalizačního zařízení, který určuje pořadí a délku signálů volno jednotlivých signálních skupin. Jedná se o cyklus. Viz obrázek 2.3.



Obrázek 2.3: Signální plán

- *Mezičas* (vyklizovací čas) - je časový interval od konce signálu volno signální skupiny po začátek signálu volno kolizní signální skupiny. V této době musí poslední (vyklizující) vozidlo projíždějící v končící době signálu volno bezpečně opustit kolizní plochu dříve, než první (najíždějící) vozidlo jedoucí v době signálu volno v kolizním směru této kolizní plochy dosáhne.
- *Tabulka mezičasů* je reprezentace matice mezičasů, kde sloupce představují najíždějící proudy a řádky vyklizující proudy. Viz tabulku 2.1.

	VA	VB	VC	VD	SC	SD	PC	PD	PE
VA			3						7
VB			10	5		5	10		
VC	6	4		7			4		
VD		5	4					4	
SC							4		
SD		1						4	
PC		0	7			7			
PD				8		8			
PE	2								

Tabulka 2.1: Tabulka mezičasů

2.2 Charakteristika úlohy

Každá světelně řízená křižovatka má svůj vlastní signální plán, podle kterého se řídí. Dle [2] se signální plán tvoří následovně:

1. Vytvoří se fázové schéma, tzn.
   - počet fází,
   - pořadí fází,
   - fázové přechody,
2. vypočítá se minimální délka jednoho cyklu,
3. určí se doba jednotlivých signálů volno.

Tato práce se zabývá prvním bodem, tj. vytvářením fázového schématu, respektive zefektivněním jeho vytváření. Další dva body už nejsou předmětem této práce.

Z podkapitoly 2.1 už je známo, že mezi jednotlivými dopravními proudy mohou existovat vztahy kolize, podmíněné kolize, nebo nekolize. Kolizní proudy nemohou mít signál volno ve stejný čas, poněvadž by se vozidla na kolizní ploše srazila. V matici mezičasů se kolizní proudy poznají podle toho, že se na místě daných dvou proudů nachází jejich mezičas. Pokud je políčko prázdné, znamená to, že jsou proudy nekolizní, nebo podmíněně kolizní. Tyto podmíněně kolizní a nekolizní proudy tvoří několik signálních fází, které pak dohromady s fázovými přechody tvoří signální plán.
V [2] se píše, že *„Počet fází vyplývá z rozdělení fází, tj. z rozhodnutí o rozčlenění dopravních pohybů na křižovatce.“* Není zde detailně popsáno, jak konkrétně fáze vytvořit. Pokud se vezme v potaz, že menší křižovatka má alespoň 4 dopravní proudy, lze vypozorovat, že u větších křižovatek existuje opravdu značné množství možných

kombinací, jakým způsobem lze proudy přiřadit do jednotlivých fází. Kdyby se vyzkoušelo jenom několik možností a první kombinace, která vyhovuje podmínkám, se uznala jako konečné řešení, mohlo by optimální řešení uniknout. Počet kombinací se sice zredukuje, jestliže se ví, že v jedné fázi mohou být současně pouze nekolizní nebo podmíněně kolizní proudy, nicméně stále hrozí zvolení řešení, které není optimální.
Co se týče pořadí fází, v [2] se píše, že *„Pořadí fází je zapotřebí navrhovat tak, aby se minimalizoval součet mezičasů."* Opět se zde nic netvrdí o tom, jak dosáhnout takového pořadí fází mezi všemi možnými kombinacemi, aby se splnila vlastnost nejmenšího součtu mezičasů.

Předpoklady

Poněvadž jde v této práci při optimalizaci fázového schématu primárně o minimalizaci mezičasů, tj. nebere se ohled na další kritéria, jako jsou doba zdržení (anglicky waiting time, dále jen WT), míra využití kapacity pruhu, počet zastavení, emise, délka front atd., je nutné tato kritéria zohlednit v předpokladech.

- Uvažuje se pouze křižovatka izolovaná a pevně řízená.
- Kolizní matice, případně potřebné údaje k jejímu dopočítání, příslušné křižovatky bude známa.
- Každý proud musí být obsažen v alespoň jedné fázi.
- Každý proud má co nejdéle zelenou a zároveň má co nejkratší červenou (aby vozidla v proudu zbytečně nečekala, což pak redukuje WT a délku front), tzn. je v co největším množství fází. Takže je zbytečné brát v potaz neúplné množiny (viz definici 3.4). Není logické nepřidat nějaký proud do fáze, pokud je to možné. Fáze se tedy uvažují pouze jako kliky (viz definici 3.5). Zohlední se tak WT i z toho pohledu, že kdyby se brala v úvahu pouze minimalizace mezičasů, snadno se najde i taková kombinace, která má celkový mezičas nulový, ale na druhé straně s nepřiměřeně velkým WT.
- V signálním plánu každý proud dostane signál volno pouze jednou, takže vozidla zbytečně vícekrát nezastavují (zohlednění kritérií počet zastavení a emise). Nevhodné varianty je možné později vyfiltrovat z výsledků při výběru optimálních fázových schémat.
- Optimální počet fází musí být dostatečně malý, neboť při menším počtu fází se v signálním plánu bude vyskytovat i méně fázových přechodů mezi jednotlivými fázemi. A jelikož po dobu trvání fázového přechodu má křižovatka nejnižší propustnost, je logicky žádoucí, aby se počet a délka fázových přechodů minimalizovaly a tím se zároveň maximalizovala propustnost křižovatky.

3 Generování fází

Tato kapitola se zabývá různými možnostmi řešení úlohy generování fází k vytvoření fázového schématu světelně řízené křižovatky. Veškeré definice, věty, obrázky a popisy algoritmů byly čerpány z [3], [4] a [5].

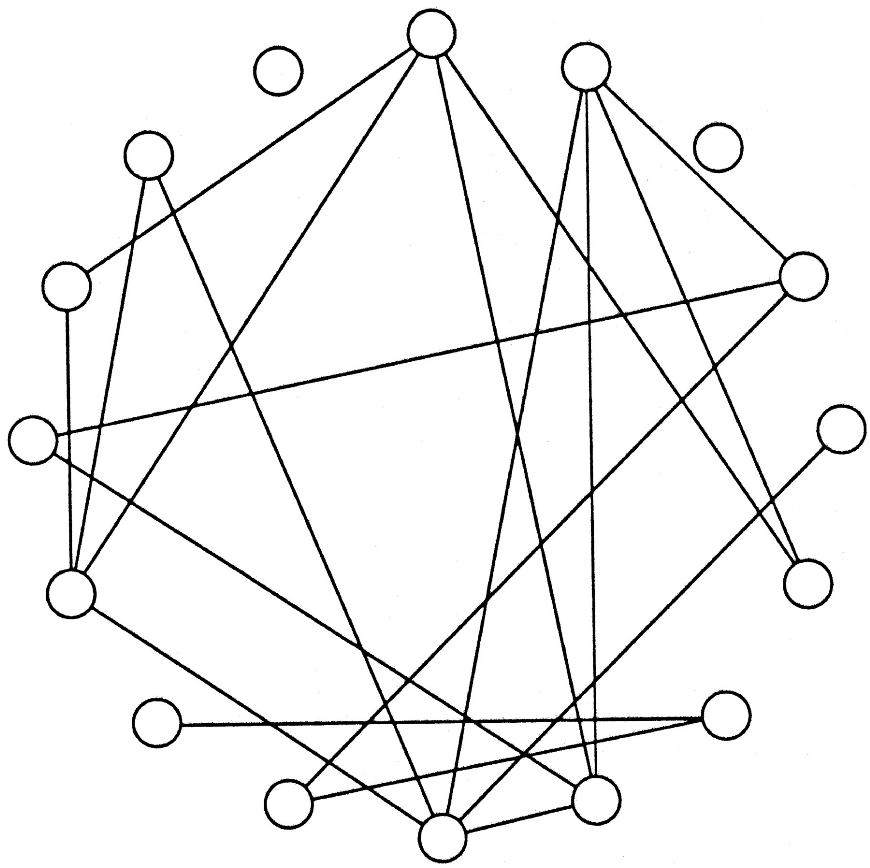
3.1 Úloha barvení grafu minimálním počtem barev

Úloha barvení grafu vznikla při tisku politických map, kdy bylo potřeba barevně odlišit státy tak, aby každé dva státy se společnou hranicí neměly stejnou barvu. Přirozeně se přitom požadovalo, aby počet barev využitý k tisku byl co nejmenší, protože použití více barev tehdy znamenalo značně vyšší finanční náklady.

Definice 3.1. *Grafem* (viz obrázek 3.1) se nazve uspořádaná dvojice $G = (V, H)$, kde $V \neq \emptyset$ je konečná množina a H je množina neuspořádaných dvojic typu $\{u, v\}$ takových, že $u, v \in V$ a $u \neq v$. Tedy

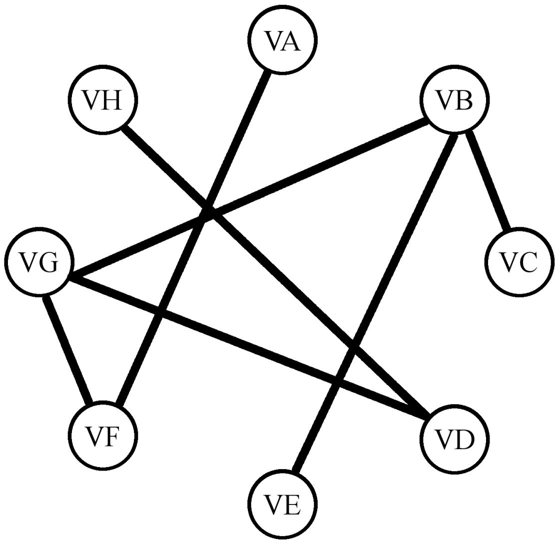
$$H \subseteq \{\{u, v\} : u \neq v, u, v \in V\} \subset V \circ V$$

Prvky množiny V se nazývají *vrcholy* a prvky množiny H *hrany* grafu G.



Obrázek 3.1: Graf
Zdroj: www.pnas.org

Úloha o barvení grafu se snaží o zabarvení všech vrcholů příslušného grafu co nejmenším počtem barev tak, aby všem vrcholům byla ve výsledku přidělena barva a žádné dva sousední vrcholy spojené hranou neměly stejnou barvu. V případě problému politických map vrcholy představují státy a sousední vrcholy jsou státy se společnou hranicí.
Úlohu o vyhledání optimálního počtu fází lze převést na úlohu barvení grafu. Podle [4] je možno tuto úlohu definovat jako *„přiřazení všech proudů do fází tak, aby počet fází byl co nejmenší, žádné dva kolizní proudy nebyly ve stejné fázi a zároveň každý proud byl přiřazen do alespoň jedné fáze“*. Platí, že vrchol představuje dopravní proud, hrana spojující dva vrcholy představuje vztah kolize mezi proudy a barva představuje fázi. Dále se tedy bude používat *graf koliznosti* (viz obrázek 3.2), který se jednoduše vytvoří pomocí tabulky mezičasů. Pokud se totiž v nějaké buňce tabulky nachází číslo, znamená to, že jsou dané dva proudy v kolizi.



Obrázek 3.2: Graf koliznosti

3.1.1 Heuristické algoritmy

Při rozsáhlejších úlohách barvení grafu nejmenším počtem barev se vyplatí použít heuristické algoritmy uvedené v této kapitole.
Mezi heuristické algoritmy řešící úlohu barvení grafu patří algoritmus sekvenčního barvení grafu, algoritmus paralelního barvení grafu a algoritmus LDF (Largest Degree First), viz [3] a [4]. Algoritmy se docela podobají, rozdíly se dají vypozorovat při vybírání vrcholů a způsobu barvení. Tyto algoritmy jsou velice jednoduché, mají ale řadu nevýhod. Když už se vrcholu přidělí nějaká barva, tato skutečnost se bere za konečnou a nelze o několik kroků později vrchol přebarvit na jinou barvu. Nejmarkantnější nevýhodou je, že se vrcholy mohou zabarvit pouze jednou barvou. V následujících částech práce jednotlivé algoritmy popíšu.

Algoritmus sekvenčního barvení

Nechť $G = (V, H)$ je graf koliznosti, kde V je množina všech vrcholů grafu G a H je množina všech hran grafu G. Dále nechť B je posloupnost barev seřazených podle jejich důležitosti, tedy $B = \{b_j\}$, kde $j = 1, 2, \ldots$. Takže barva b_1 je nejdůležitější, barva b_2 druhá nejdůležitější atd.

Algoritmus vypadá následovně:

1. Zvolí se zcela libovolná posloupnost vrcholů $P = \{v_i\}$ grafu G, kde $i = 1, 2, ..., n$.
2. Postupně se prochází posloupnost P a barví se vrchol v_i barvou s nejmenším indexem takovou, kterou nejsou obarveni sousedé vrcholu v_i. Až se všechny vrcholy obarví, skončí se.

Sousedy vrcholu v_i se rozumí vrcholy spojené s vrcholem v_i.
V tomto algoritmu se tedy postupně vybírají všechny vrcholy a barví se barvou s nejnižším indexem takovou, kterou nejsou obarveni sousedé tohoto vrcholu.

Algoritmus paralelního barvení

Zatímco předchozí algoritmus vybírá vrcholy z posloupnosti a barví je barvou s nejnižším indexem, algoritmus paralelního barvení grafu vybere barvu s nejnižším indexem a pomocí této barvy obarví co nejvíce vrcholů grafu, samozřejmě s dodržením pravidla nestejnosti barev sousedních vrcholů. Až tuto barvu vyčerpá, vybere další barvu a opět barví co největší možný počet vrcholů. Tyto kroky algoritmus opakuje, až jsou všechny vrcholy obarveny.
Další věcí, kterou se tento algoritmus liší od předešlého, je posloupnost vrcholů. Tentokrát není zcela libovolná, ale vrcholy se seřadí sestupně podle svých stupňů.

Definice 3.2. *Stupeň vrcholu v* grafu G je počet hran zasahujících do vrcholu v.

Nechť $G = (V, H)$ je graf koliznosti, kde V je množina všech vrcholů grafu G a H je množina všech hran grafu G. Nechť $B = \{b_j\}$, kde $j = 1, 2, ...$, je posloupnost barev seřazených podle jejich důležitosti.

Algoritmus pak postupuje takto:

1. Zvolí se posloupnost vrcholů $P = \{v_i\}$, kde $i = 1, 2, ..., n$, seřazených sestupně podle jejich stupně, tedy p_1 je vrchol s nejvyšším stupněm.
2. Postupně se projde celá posloupnost P: Jestliže vrchol v_i není obarven barvou b_j a žádný soused není zabarvený barvou b_j, pak se vrchol v_i obarví barvou b_j.
3. Pokud jsou všechny vrcholy posloupnosti P zabarvené, algoritmus skončí. Jestliže nejsou všechny vrcholy posloupnosti zabarvené, $j = j + 1$ a přejde se na druhý bod algoritmu.

Algoritmus LDF (Largest Degree First)

Jedná se v podstatě o rozšířený sekvenční algoritmus. Oproti klasickému sekvenčnímu algoritmu, algoritmus LDF nemá náhodně zvolenou posloupnost vrcholů, nýbrž si za pochodu volí, jaký vrchol se nabarví barvou s nejnižším indexem.

Definice 3.3. *Barevný stupeň* vrcholu v je celkový počet různých barev, kterými jsou zabarvené sousední vrcholy vrcholu v.

Nechť je $G = (V, H)$ graf koliznosti a $B = \{b_j\}$, kde $j = 1, 2, ...$ je posloupnost barev seřazených podle jejich důležitosti.

Algoritmus LDF:

1. Zvolí se posloupnost vrcholů $P = \{v_i\}$, kde $i = 1, 2, ..., n$, seřazených sestupně podle jejich stupně.
2. Vyberou se všechny nezabarvené vrcholy v_i s největším stupněm a z nich se vybere vrchol s největším barevným stupněm.
3. Tento vrchol se obarví barvou s nejmenším možným indexem.
4. Pokud se všechny vrcholy obarvily, algoritmus končí. V opačném případě se přejde na druhý bod algoritmu.

3.1.2 Matematický model

Výše zmíněné heuristické algoritmy ve většině případů nenajdou celkové optimální řešení. Navíc jsou algoritmy postaveny tak, aby každému vrcholu určily pouze jednu barvu, respektive každý dopravní proud zařadily jen do jedné fáze. Což ale opomíná předpoklad, aby se každý proud zařadil do co nejvíce fází. Tento předpoklad oproti heuristickým algoritmům zohledňuje matematický model, který dokáže vrchol obarvit více barvami, respektive dokáže přiřadit proud do více fází, pokud je to možné.
Na následujících řádcích popíšu matematický model, jež lze použít k řešení úloh barvení grafu.

Nechť $G = (V, H)$ je graf koliznosti, kde V je množina všech vrcholů grafu G a H je množina všech hran grafu G. Dále nechť B je nejmenší množina potřebných barev na barvení vrcholů grafu a y je mohutnost B. Nechť A je incidenční matice velikosti n x n s prvky $a_{vw} \in A$, kde n je mohutnost V a $v, w \in V$. Tato matice říká, jestli jsou dva vrcholy sousední ($a_{vw} = 1$), nebo ne ($a_{vw} = 0$). Dále nechť X je matice velikosti n x y, která vyjadřuje, jakými barvami b je ($x_{vb} = 1$) a není ($x_{vb} = 0$) vrchol v obarven, kde $b \in B$. Pak se matematický model definuje následovně.

Účelová funkce:

$$max \sum_{v \in V} \sum_{b \in B} x_{vb} \tag{3.1}$$

Podmínky:

$$\sum_{b \in B} x_{vb} \geq 1 \qquad v \in V \tag{3.2}$$

$$x_{vb} + x_{wb} \leq 1 \qquad v \in V, b \in B, v \neq w, a_{vw} = 1 \tag{3.3}$$

$$x_{vb} \in \{0, 1\} \qquad v \in V, b \in B \tag{3.4}$$

Účelová funkce (3.1) maximalizuje počet vrcholů v obarvených barvou b. Podmínka (3.2) říká, že každý vrchol musí být obarven alespoň jednou barvou a zároveň nevylučuje obarvenost více barvami. Pomocí (3.3) se zaručí, aby každé sousední vrcholy neměly stejnou barvu. Podmínka (3.4) reprezentuje binární proměnnou.

Jak se zjistí, kolik je nejmenší potřebný počet barev k barvení? Buď vylučovacím způsobem - proudy se zkouší rozdělit nejprve do 1 fáze, což nikdy nejde, přejde se tedy ke 2 fázím atd. nebo lze použít matematický minimalizační model [4].

3.2 Pokrytí grafu nejmenším počtem klik

Nejprve je potřeba si přesněji definovat pojmy, které se v této podkapitole používají.

Definice 3.4. Nechť $K \subseteq V$. Pak K je *úplná množina v grafu* G, jestliže každé dva různé prvky z množiny K spolu sousedí (jsou spojeny hranou).

Definice 3.5. $K \subseteq V$ je *maximální úplná množina* nebo také *klika v grafu* G, pokud je K úplná v G a zároveň K není podmnožinou žádné jiné úplné množiny v G.

S požadavkem zařadit každý proud do co nejvíce fází se počítalo již v matematickém modelu. K vyhledání minimálního počtu fází lze použít mimo úlohu barvení grafu i úlohu pokrytí grafu nejmenším počtem maximálních úplných množin, která je přímo založena na tomto požadavku. Konkrétně se hledají maximální úplné množiny v grafu nekoliznosti, kde množiny ztělesňují fáze, vrcholy opět představují proudy křižovatky a hrany reprezentují vztah nekolize nebo podmíněné kolize mezi proudy.
Úloha se dá řešit dvěma způsoby. První způsob: Nejdříve se graf nekoliznosti zabarví nejmenším počtem barev (viz podkapitolu 3.1), tím se vytvoří úplné množiny se stejně zabarvenými vrcholy. Potom už stačí tyto množiny rozšířit na maximální. Tento způsob je zahrnut pod matematickým modelem s tím, že matematický model je univerzálnější, protože se neomezuje pouhou jednou kombinací prvotních úplných množin. Prvnímu způsobu se tedy nebude věnovat zvláštní pozornost.
Druhý způsob: Nejprve se hledají všechny maximální úplné množiny - k tomu je možné použít algoritmus na vyhledání všech klik v grafu nekoliznosti, nebo se dají najít i ručně - a v dalším kroku se selektuje taková kombinace s nejmenším počtem těchto množin, která obsahuje všechny vrcholy - lze tady použít algoritmus na řešení pokrývacího problému. V následujícím textu popíšu druhý způsob.
Existuje ale i jiné východisko. Všechny maximální úplné množiny lze najít i ručně bez nějakého promyšleného algoritmu. Pokud je graf nekoliznosti opravdu řídký, připadá tato možnost v úvahu. Stačí se podívat na graf a hledat všechny největší množiny splňující definici úplnosti. Jestli je ale graf moc hustý, tento postup nepřipadá v úvahu.

3.2.1 Exaktní algoritmus na hledání všech klik v grafu

Metoda vyhledání všech klik grafu spočívá v tom, že se prohledají všechny možnosti. Nechť $V = \{1, 2, ..., n\}$ je množina všech vrcholů a $K_i = \{K(1), K(2), ..., K(i)\}$ je pomocná množina vrcholů, jejíž obsah se dynamicky mění podle toho, jak se bude procházet algoritmus. Bude tvořit k-prvkové úplné podmnožiny a vrcholy v ní budou někdy ubývat a někdy přibývat. Její podoby se dají chápat jako tvořící se kliky. Dále nechť V_x je množina všech sousedících vrcholů vrcholu x. V průběhu algoritmu se bude počítat i s dalšími dvěma množinami vrcholů $C(i)$ a $N(i)$, pro které platí:

- Všechny vrcholy $v \in C(i) \cup N(i)$ se dají přidat k pomocné množině K_i, tj. pro všechny vrcholy $v \in C(i) \cup N(i)$ je $K_i \cup \{v\}$ úplnou množinou.

- $C(i)$ je množina všech vrcholů, které se ještě neobjevily v K_i, tj. žádná úplná množina typu $K_i \cup \{v\}$ ještě nebyla prozkoumaná pro žádné $v \in C(i)$.
- $N(i)$ je množina všech vrcholů v, u kterých se už vyzkoušelo přidání ke K_i, tj. všechny kliky obsahující $K_i \cup \{v\}$, kde $v \in N(i)$, už se našly.
- $(C(i) \cup N(i)) \cap K_i = \emptyset$.

Algoritmus na vyhledání všech klik grafu pak vypadá takto:

1. $k = 0$, $N(0) = \emptyset$, $C(0) = V$,
2. $x = min\{C(k)\}$, $K(k+1) = x$, $N(k+1) = N(k) \cap V_x$, $C(k+1) = C(k) \cap V_x$, $k = k+1$,
3. jestli $C(k) = \emptyset$ a zároveň $N(k) = \emptyset$, $K_k = \{K(1), K(2), ..., K(k)\}$ je klika a přejdi na bod 6,
4. pokud $C(k) = \emptyset$, přejdi na bod 6,
5. existuje-li vrchol $p \in N(k)$, který sousedí se všemi vrcholy z $C(k)$, přejdi na další bod, v opačném případě se vrať na bod 2,
6. jestliže $k = 0$, skonči, jinak $x = K(k)$, $k = k - 1$, $C(k) = C(k) - \{x\}$, $N(k) = N(k) \cup \{x\}$ a přejdi na bod 4.

V prvním bodě se iniciují počáteční hodnoty. V druhém bodě, pokud už je minimálně jeden vrchol v K_k, se vybere sousedící vrchol s nejnižší hodnotou, přidá se do K_{k+1} a proběhne aktualizace množin. Ve třetím bodě se testuje maximálnost K_k, tj. tady se případně objeví nová klika. Ve čtvrtém bodě se testuje možnost zvětšení K_k. V pátém bodě se testuje předčasné ukončení rozšiřování K_k, aby se zbytečně množina nerozšiřovala o vrchol, který už se vyzkoušel, a následně by algoritmus zjistil, že se musí o krok vrátit. V posledním bodě se algoritmus vrací do stavu před přidáním posledního vrcholu (kromě $N(k)$).

3.2.2 Model pokrývacího problému

K dispozici jsou kliky grafu nekoliznosti. Na řadu přichází vyhledání kombinace s nejmenším počtem těchto množin, která současně pokrývá všechny vrcholy. Na tuto úlohu se aplikuje matematický model řešící pokrývací problém.

Nechť $C = (c_{ij})$ je matice typu n x m, kde n je počet vrcholů grafu a m je počet klik. Prvky c_{ij} se rovnají jedné, jestliže klika j, obsahuje vrchol i. V opačném případě $c_{ij} = 0$. Cílem je vybrat nejmenší počet sloupců matice C takovým způsobem, aby se minimálně jedna jednička z každého řádku vyskytla alespoň v jednom z vybraných sloupců. Dále nechť S je množina vybraných sloupců a $x_j \in \{0, 1\}$. $x_j = 1$, když se vybere sloupec j do množiny S, v opačném případě $x_j = 0$. Pak matematický model vypadá následovně.

Účelová funkce:

$$min \sum_{j=1}^{m} x_j \tag{3.5}$$

Podmínky:

$$\sum_{j=1}^{m} c_{ij} x_j \geq 1 \qquad \forall i \in \{1, 2, ..., n\} \tag{3.6}$$

$$x_j \in \{0, 1\} \qquad \forall j \in \{1, 2, ..., m\} \tag{3.7}$$

V (3.5) se model snaží najít vektor $\mathbf{x} = (x_1, x_2, ..., x_m)$ o nejméně složkách, který značí, kolik se vybralo sloupců. Z první podmínky jde vidět, že $c_{ij}x_j = 1$, pokud je ve sloupci j a řádku i číslo 1. Také říká, že alespoň jedna jednička z každého řádku bude v nějakém vybraném sloupci. (3.7) vyjadřuje binární proměnnou.

3.2.3 Heuristický algoritmus

[5] nabízí k řešení předchozího problému jednoduchý heuristický algoritmus, u kterého se ještě nestalo, že by *„nevedl k optimálnímu výsledku v tom smyslu, že by vybral více fázových skupin, než je optimální počet“*. Navíc se takto najde více přípustných řešení.

Nechť $C = (c_{ij})$ je pokrývací matice typu n x m z předešlé podkapitoly. Opět je snaha v matici C najít nejmenší počet sloupců obsahující alespoň jednu jedničku z každého řádku. Pak se algoritmus skládá z kroků:

1. Má-li nějaký řádek matice C jedinou jedničku, pak se sloupec obsahující tuto jedničku přesune do konečného řešení. Poté se tento sloupec a všechny řádky, ve kterých je ve vybraném sloupci jednička, vyřadí z matice.
2. Jestliže je nějaký sloupec a podmnožinou sloupce b, sloupec a se vyřadí z matice.
3. Pokud $C \neq \emptyset$ a lze provést krok 1 nebo 2, provedou se. Když ani jeden z kroků nelze provést, vybere se x-tý sloupec z nynější matice a přesune se do konečného řešení. Dále se tento sloupec a řádky obsahující jeho jedničky odeberou z matice. Jestli $P = \emptyset$, algoritmus končí, v opačném případě se přejde na krok 1.

Poznámka: *x-tým sloupcem* se myslí, aby se vyzkoušely všechny možnosti, tj. nejprve se vybere první sloupec a pokračuje se v algoritmu. Až algoritmus skončí, vrátí se do stavu, kdy se vybral první sloupec a tentokrát se vybere druhý sloupec. Takhle se pokračuje, dokud se nevyzkouší všechny možnosti.
V případě, že bylo obdrženo více možných výsledků, se může jejich optimálnost porovnat tak, že se na všechny možnosti aplikují algoritmy z následující kapitoly.

4 Optimální pořadí fází

Předchozí kapitola se zabývala určením minimálního počtu fází a přiřazením proudů do těchto fází, v této kapitole se nyní ukáže, jak se tyto fáze seřadí. Jak již bylo řečeno, mezi jednotlivými fázemi probíhají fázové přechody skládající se alespoň z příslušných mezičasů. Mezičasy jsou neefektivní částí signálního plánu, je tedy potřeba fázové přechody navrhnout minimalisticky. Pro různá pořadí fází vyplývají různé hodnoty mezičasů, proto se hledá optimální pořadí fází.
V následujícím textu budou popsány dvě metody, kterými se dá tato úloha řešit - heuristický algoritmus a model řešící úlohu obchodního cestujícího. Obě metody jsou čerpány z [6].

4.1 Heuristický algoritmus

Heuristický algoritmus je jednoduchý a určený spíše k „ručnímu“ propočítání. Postup vypadá následovně.

1. Sestaví se všechny možné kombinace pořadí fází.
2. Určí se všechny mezičasy mezi fázemi v uvažovaných kombinacích (viz tabulku 4.1). To se pro každou dvojici fází provede tak, že se v tabulce mezičasů najdou všechny proudy z vyklizující fáze a z najíždějící fáze, udělá se jejich průnik a vybere se z nich nejvyšší mezičas.

	VA1	VA2	VA3	VB1	VB2	VB3	VC1	VC2	VD1
VA1			3				4		7
VA2			7	5		5	10		
VA3	6	4		5			3		
VB1		11	3					4	
VB2							4		
VB3		1						4	
VC1	4	0	6			7			
VC2				8		8			
VD1	3								

Tabulka 4.1: Tabulka mezičasů

3. Pro každou sestavenou kombinaci se vypočítá její celkový mezičas tak, že se sečtou všechny určené mezičasy mezi fázemi dané kombinace.
4. Dále se určí ze všech celkových mezičasů ten nejmenší. Kombinace s vybraným celkovým mezičasem se prohlásí za optimální pořadí fází.

První krok je opravdu důležitý, poněvadž pokud se nenajdou všechny možnosti, nemusí se optimální kombinace nalézt.

4.2 Model obchodního cestujícího

Dalším prostředkem, pomocí kterého je možné vyhledat optimální pořadí fází, je úloha obchodního cestujícího. V této úloze se obchodní cestující snaží projít všechna daná města právě jedenkrát, přitom urazit co nejkratší vzdálenost a na konci se vrátit do počátečního města.
Úloha najít optimální pořadí fází lze převést na úlohu obchodního cestujícího tak, že města budou představovat fáze, vzdálenost mezi dvěma městy bude maximální mezičas mezi dvěma fázemi a celková uražená vzdálenost cestujícího bude reprezentovat výsledný celkový mezičas. Cílem je tedy najít řešení, kdy každá fáze proběhne právě jednou a výsledný celkový mezičas bude minimální.

Nechť $F = \{1, 2..., N\}$ je množina fází a $i, j \in F$. Dále buď $x_{ij} \in \{0, 1\}$ proměnná charakterizující vztah mezi fázemi i a j, tedy pokud po fázi i následuje fáze j, platí $x_{ij} = 1$, v opačném případě se položí $x_{ij} = 0$. Nechť $C = (c_{ij})$ je matice mezičasů, kde c_{ij} jsou mezičasy mezi fázemi i a j (viz bod 2 v heuristickém algoritmu). Nechť y_i vyjadřuje pozici fáze i, na které se daná fáze i realizovala. Potom model vypadá takto.

Účelová funkce:

$$min \sum_{i\in F}\sum_{j\in F} c_{ij}x_{ij} \qquad i \neq j \tag{4.1}$$

Podmínky:

$$\sum_{i\in F} x_{ij} = 1 \qquad j \in F, i \neq j \tag{4.2}$$

$$\sum_{j\in F} x_{ij} = 1 \qquad i \in F, i \neq j \tag{4.3}$$

$$y_i - y_j + N x_{ij} \leq N - 1 \qquad i, j = 2, ..., N; i \neq j \tag{4.4}$$

$$x_{ij} \in \{0, 1\} \qquad i, j \in F, i \neq j \tag{4.5}$$

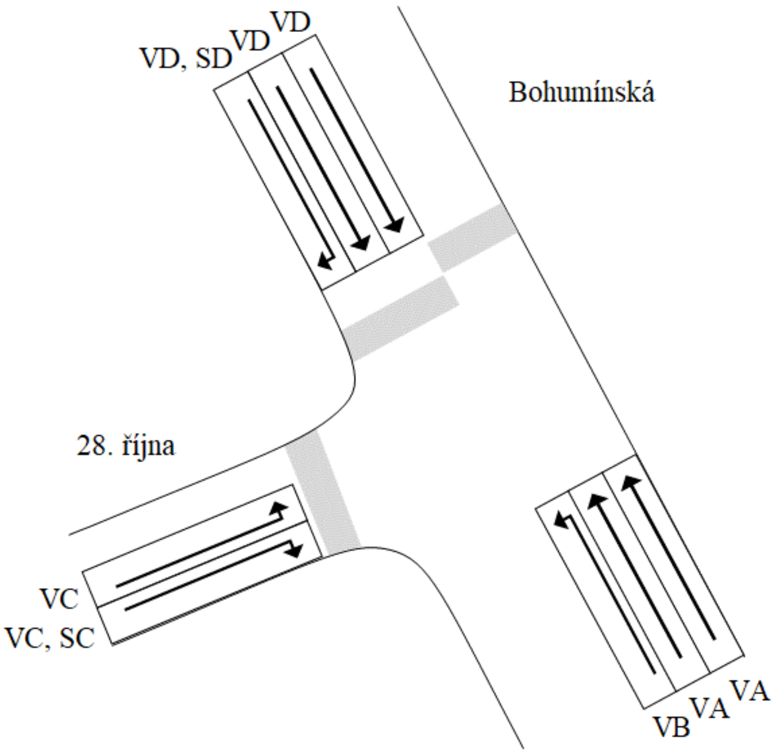
$$y_i \geq 0 \qquad i = 2, ..., N \tag{4.6}$$

V (4.1) se minimalizuje celkový mezičas, podmínky (4.2) a (4.3) zaručují, že každá fáze proběhne právě jednou. (4.4) eliminuje podcykly a poslední dvě podmínky jsou nutnými podmínkami modelu.

5 Případová studie

V této kapitole bude uveden model křižovatky, se kterým se tu dále pracuje. Dopravní proudy příslušné křižovatky se přiřadí do fází pomocí metod zmíněných v teoretické části. Dále se na všechny získané kombinace fází, aplikují nejprve ruční heuristický algoritmus, následně i program řešící úlohu obchodního cestujícího. Poslední část se zabývá finálním sestavením signálního plánu nalezeného řešení.

5.1 Model křižovatky



Obrázek 5.1: Model křižovatky

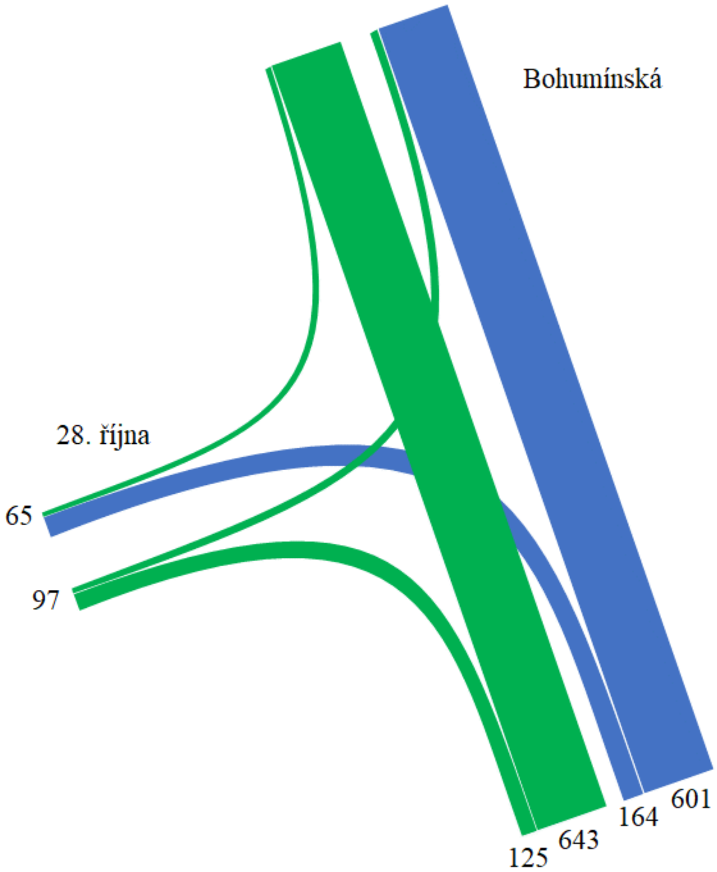
Jako model byla vybrána křižovatka (viz obrázek 5.1) v Ostravě mezi ulicemi Bohumínská a 28. října, byla čerpána z [7]. Křižovatka má celkem 9 dopravních proudů, respektive signálů: VA, VB, VC, VD, SC, SD, PC, PD, PE. Signály označené písmenem

- V značí plné signály pro vozidla,
- S značí signály s doplňkovou zelenou šipku pro vozidla,
- P značí signály pro chodce.

Z obrázku 5.2 lze vyčíst intenzity jednotlivých proudů, níže jsou intenzity uvedeny v jednotkách [vozidlo/hodina]:

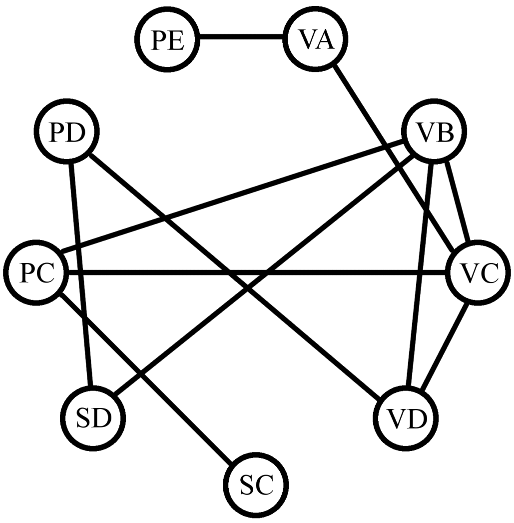
- plné signály pro vozidla VA: 601, VB: 164, VC: 97, VD: 643,
- směrové šipky SC: 125, SD: 65,
- přechody pro chodce PC, PD, PE: neznámé.

Lze tedy říci, že proudy VA a VB na ulici Bohumínská jsou proudy s největšími intenzitami a že ostatní proudy nejsou tak frekventované.



Obrázek 5.2: Intenzity jednotlivých proudů

Později bude potřeba znát i graf koliznosti (viz obrázek 5.3), tabulku nekoliznosti (viz tabulka 5.1) a tabulku mezičasů křižovatky (viz tabulka 5.6).



Obrázek 5.3: Graf koliznosti

5.2 Generování fází

Hledání všech klik

V předpokladech bylo uvedeno, že by každý dopravní proud měl být obsažen v co nejvíce fázích, tzn. výsledné fáze budou maximální úplné množiny - kliky. Nejprve se tedy najdou všechny kliky, které se dále označí a budou se pomocí nich zapisovat výsledky všech následujících algoritmů.
K algoritmu na hledání všech klik byl použit skript naprogramovaný v Matlabu [8]. Skript je uveden i v příloze na CD. Ke vstupu je potřeba zadat tabulku nekoliznosti 5.1. Vytvoří se tak, že se vezme tabulka mezičasů a všechna čísla se přepíší na nuly. Na diagonálu se také napíší nuly a do zbylých prázdných míst se napíší jedničky. Jedničky a nuly v tabulce značí vztah nekoliznosti, respektive podmíněné koliznosti mezi dvěma proudy. Pokud jsou dané proudy i a j nekolizní, respektive podmíněně kolizní, v tabulce se pak na pozici i, j nachází 1. Jestliže jsou kolizní, políčko je vyplněno nulou. Na diagonále se nuly vyskytují, neboť tak vyžaduje výše zmíněný skript.

	VA	VB	VC	VD	SC	SD	PC	PD	PE
VA	0	1	0	1	1	1	1	1	0
VB	1	0	0	0	1	0	0	1	1
VC	0	0	0	0	1	1	0	1	1
VD	1	0	0	0	1	1	1	0	1
SC	1	1	1	1	0	1	0	1	1
SD	1	0	1	1	1	0	1	0	1
PC	1	0	0	1	0	1	0	1	1
PD	1	1	1	0	1	0	1	0	1
PE	0	1	1	1	1	1	1	1	0

Tabulka 5.1: Tabulka nekoliznosti

Algoritmus na hledání všech klik našel celkem 10 klik, které jsou znázorněny v tabulce 5.2, kde se v sloupcích nacházejí kliky (označeny číselně od 1 do 10) a v řádcích jsou dopravní proudy. Na pozici i, j se pak nachází 1, pokud klika j obsahuje daný proud i.

	Kliky									
	1	2	3	4	5	6	7	8	9	10
VA	1	1	0	0	0	0	1	1	0	0
VB	1	0	1	0	0	0	0	0	0	0
VC	0	0	0	1	1	0	0	0	0	0
VD	0	1	0	0	0	1	1	0	1	0
SC	1	1	1	1	1	1	0	0	0	0
SD	0	1	0	1	0	1	1	0	1	0
PC	0	0	0	0	0	0	1	1	1	1
PD	1	0	1	0	1	0	0	1	0	1
PE	0	0	1	1	1	1	0	0	1	1

Tabulka 5.2: Kliky

Sekvenční barvení

Posloupnost P se volí libovolně.

$$P = \{VA, VB, VC, VD, SC, SD, PC, PD, PE\}$$

A vrcholy se postupně začnou barvit (červená barva značí barvu 1, oranžová barvu 2 a modrá barvu 3). Vrchol VA se obarví barvou 1, vrchol VB také barvou 1, protože není spojen v grafu koliznosti 5.3 hranou s vrcholem VA atd.

Obarvení vrcholů pak vypadá následovně:

1. barva: VA, VB, SC, PD,
2. barva: VC, SD, PE,
3. barva: VD, PC.

VA, VB, VC, VD, SC, SD, PC, PD, PE

Barvy se nyní přepíší na fáze:

1. fáze: VA, VB, SC, PD,
2. fáze: VC, SD, PE,
3. fáze: VD, PC.

Fáze se poté doplní do úplných maximálních množin a vznikne jejich výsledná podoba:

1. fáze = klika 1: VA, VB, SC, PD,
2. fáze = klika 4: VC, SD, PE, SC,
3. fáze = klika 7: VD, PC, VA, SD.

Sekvenční barvení upravené

Tento algoritmus pracuje stejně jako sekvenční barvení s výjimkou toho, že se zde pracuje s nelibovolnou posloupností vrcholů. Posloupnost dopravních proudů se v této variantě vytvořila tak, že se proudy seřadily podle svých intenzit od nejzatíženějších po nejméně frekventované, přechody pro chodce jsou seřazeny libovolně, viz obrázek 5.2.

$$P = \{VD, VA, VB, SC, VC, SD, PC, PE, PD\}$$

Nyní se aplikuje algoritmus sekvenčního barvení (objeví se tu čtvrtá barva - šedá).

VD, VA, VB, SC, VC, SD, PC, PE, PD

Barvy se přepíší na fáze a ty se doplní na kliky.

1. fáze = klika 2: VA, VD, SC, SD,
2. fáze = klika 3: VB, PE, OD, SC,
3. fáze = klika 4: VC, SC, SD, PE,
4. fáze = klika 7: PC, VA, VD, SD.

Paralelní barvení

Vrcholy se seřadí podle svých stupňů a pokud mají 2 vrcholy stejný stupeň, jejich pořadí se volí libovolně.

$$P = \{VC, VB, PC, VD, SD, VA, PD, PE, SC\}$$

Nyní se začne barvit tak, že se vezme jedna barva, s ní se obarví popořadě co nejvíce vrcholů, dále se vezme další barva atd.

VC, VB, PC, VD, SD, VA, PD, PE, SC

↓

VC, VB, PC, VD, SD, VA, PD, PE, SC

↓

VC, VB, PC, VD, SD, VA, PD, PE, SC

↓

VC, VB, PC, VD, SD, VA, PD, PE, SC

Výsledné obarvení vrcholů tedy vypadá (vrcholy se přiřazují přímo do fází):

1. fáze: VC, SD, PE, SC,
2. fáze: VA, VB, PD,
3. fáze: PC, VD.

Fáze se pak doplní do úplných maximálních množin.

1. fáze = klika 4: VC, SD, PE, SC,
2. fáze = klika 1: VA, VB, PD, SC,
3. fáze = klika 7: PC, VD, VA, SD.

Barvení LDF

Vrcholy se seřadí podle svých stupňů a pokud mají 2 vrcholy stejný stupeň, seřadí se podle svých barevných stupňů. A jestliže jsou 2 vrcholy stejného stupně a zároveň i stejného barevného stupně, seřadí se libovolně:

$$P = \{VC, VB, PC, VD, PD, SD, VA, SC, PE\}$$

Nyní se vyberou nezabarvené vrcholy s největším stupněm - VC, VB. Jelikož mají oba dva vrcholy nulový barevný stupeň, tj. identický, zvolí se třeba VC a obarví se jako první. VB se obarví jako druhý. Následně se opět vyberou všechny nezabarvené vrcholy s největším stupněm - PC, VD - a algoritmus pokračuje obdobně.

VC, VB, PC, VD, PD, SD, VA, SC, PE

↓

VC, VB, PC, VD, PD, SD, VA, SC, PE

↓

VC, VB, PC, VD, PD, SD, VA, SC, PE

↓

VC, VB, PC, VD, PD, SD, VA, SC, PE

↓

VC, VB, PC, VD, PD, SD, VA, SC, PE

Výsledné obarvení vrcholů vypadá následovně (vrcholy jsou přiřazeny přímo do fází):

1. fáze: VC, PD, SC, PE,
2. fáze: VB, VA,
3. fáze: PC, VD, SD.

A po doplnění na úplné maximální množiny.

1. fáze = klika 5: VC, PD, SC, PE,
2. fáze = klika 1: VB, VA, SD, PD,
3. fáze = klika 7: PC, VD, SD, VA.

Matematický model

Všechny proudy nelze rozdělit do dvou fází a některým barvícím algoritmům se podařilo vytvořit 3 fáze. Což znamená, že nejmenší možný počet fází je 3. S tím lze pracovat dále v matematickém modelu. Model je naprogramován v modelovacím prostředí GAMS, viz přílohu na CD. Matematický model vygeneroval tyto fáze:

1. fáze = klika 1: VA, VB, SC, PD,
2. fáze = klika 9: VC, SC, PD, PE,
3. fáze = klika 5: VD, SD, PC, PE.

Pokrývací problém

Jelikož už jsou vyhledané všechny kliky, lze aplikovat model řešící pokrývací problém. Model je, stejně jako předchozí matematický model, naprogramován v GAMS, viz přílohu na CD. Pokrývací algoritmus našel tyto fáze:

1. fáze = klika 1: VA, VB, SC, PD,
2. fáze = klika 5: VC, SC, PD, PE,
3. fáze = klika 7: VA, VD, SD, PC.

Heuristický algoritmus

V heuristickém algoritmu se pracuje pouze s tabulkou 5.2.
Nelze provést krok 1 ani 2, takže se vybere klika 1 a přesune se do konečného řešení. Následně se vyřadí řádky, ve kterých jsou v daném sloupci jedničky - VA, VB, SC, PD. Což vede k tabulce 5.3.

	Kliky								
	2	3	4	5	6	7	8	9	10
VC	0	0	1	1	0	0	0	0	0
VD	1	0	0	0	1	1	0	1	0
SD	1	0	1	0	1	1	0	1	0
PC	0	0	0	0	0	1	1	1	1
PE	0	1	1	1	1	0	0	1	1

Tabulka 5.3: Zredukovaná tabulka 1

Nyní se provede krok 2 - sloupce 2, 3, 5, 6, 7, 8, 10 jsou podmnožinou některých jiných sloupců, takže se z tabulky vyřadí. Obdrží se tak tabulka 5.4.

	Kliky	
	4	9
VC	1	0
VD	0	1
SD	1	1
PC	0	1
PE	1	1

Tabulka 5.4: Zredukovaná tabulka 2

Řádky VC a VD mají každý jen jednu jedničku, takže se obě kliky 4 a 9 přesunou do konečného řešení. Našlo se tak první řešení heuristického algoritmu: kliky 1, 4, 9.
Nyní se algoritmus vrátí před okamžik, kdy na začátku vybral do konečného řešení kliku 1, a místo ní se vybere klika 2. Analogicky probíhá postup i s klikou 2. Celý algoritmus skončí, až se vybere i poslední klika 10. Výsledek heuristického algoritmu je znázorněn v tabulce 5.5, kde jednotlivá čísla představují dané kliky.

	Fáze 1	Fáze 2	Fáze 3	Fáze 4
Různá řešení	1	4	9	
	1	5	7	
	1	5	9	
	3	4	7	
	3	5	7	
	1	2	4	10
	1	5	6	8
	2	3	5	10
	3	4	6	8

Tabulka 5.5: Řešení heuristického algoritmu

5.3 Optimální pořadí fází

	VA	VB	VC	VD	SC	SD	PC	PD	PE
VA			3						7
VB			10	5		5	10		
VC	6	4		7			4		
VD		5	4					4	
SC							4		
SD		1						4	
PC		0	7		7				
PD				8		8			
PE	2								

Tabulka 5.6: Tabulka mezičasů

K optimálnímu seřazení fází je potřeba znát tabulku mezičasů dané křižovatky, v tomto případě je to tabulka 5.6.

Heuristický algoritmus

Postup algoritmu se ukáže na výstupu sekvenčního barvení, ostatní výsledky zbylých algoritmů se získají analogicky. Výstupem sekvenčního barvení jsou kliky 1, 4 a 7. V úvahu tedy připadají 2 kombinace: 1-4-7 a 1-7-4. Vybere se třeba 1-4-7. Nyní se zjistí, jaké proudy obsahují kliky 1 (vyklizující) a 4 (najíždějící), a v tabulce mezičasů se označí - vyklizující modře, najíždějící zeleně (viz tabulku 5.7). Z jejich průniku se vybere největší mezičas, kterým je 10 sekund. To stejné se provede u dvojic 4-7 a 7-1, získanými mezičasy jsou 7 a 7. Celkový mezičas varianty 1-4-7 je pak $10 + 7 + 7 = 24$ sekund.
V tabulce 5.9 lze vidět všechna nalezená řešení pomocí heuristického algoritmu.

	VA	VB	VC	VD	SC	SD	PC	PD	PE
VA			3						7
VB			10	5		5	10		
VC	6	4		7			4		
VD		5	4					4	
SC							4		
SD		1						4	
PC		0	7			7			
PD				8		8			
PE	2								

Tabulka 5.7: Největší mezičas

Model obchodního cestujícího

K řešení úlohy obchodního cestujícího byl použit skript naprogramovaný v Matlabu [9]. Skript je uveden i v příloze na CD. Ke vstupu je potřeba zadat tabulku mezičasů příslušných klik. Pro variantu klik 1-4-7 je to tabulka 5.8. Vytvoří se tak, že se vypočítají mezičasy mezi jednotlivými klikami, analogicky, jak se počítalo v předchozím heuristickém algoritmu.

	1	4	7
1	0	10	10
4	6	0	7
7	7	7	0

Tabulka 5.8: Tabulka mezičasů klik 1-4-7

Program řešící úlohu obchodního cestujícího má výsledky identické jako heuristický algoritmus (viz tabulku 5.9) kromě dvou řešení 1-6-8-5 a 2-5-10-3 - tato řešení nenašel.
Z tabulky 5.9 jde vidět, že existují 2 optimální řešení signálního plánu, obě dvě se součtem mezičasů 21 sekund: 3-7-4 a 3-7-5.

	Fáze				Pořadí klik				Součet mezičasů [s]
	1	2	3	4					
Sekvenční barvení	1	4	7		1	7	4		23
Sekvenční upravené	2	3	4	7	2	7	4	3	23
Paralelní barvení	4	1	7		1	7	4		23
Algoritmus LDF	5	1	7		1	7	5		23
Matematický model	1	9	5		1	9	5		23
Pokrývací algoritmus	1	5	7		1	7	5		23
	1	4	9		1	9	4		23
	1	5	7		1	7	5		23
	1	5	9		1	9	5		23
	3	4	7		3	7	4		21
Heuristický algoritmus	3	5	7		3	7	5		21
	1	2	4	10	1	2	4	10	26
	1	5	6	7	1	6	5	8	25
	1	5	6	7	1	6	8	5	25
	2	3	5	10	2	10	5	3	26
	2	3	5	10	2	5	10	3	26
	3	4	6	8	3	6	8	4	24

Tabulka 5.9: Optimální pořadí fází

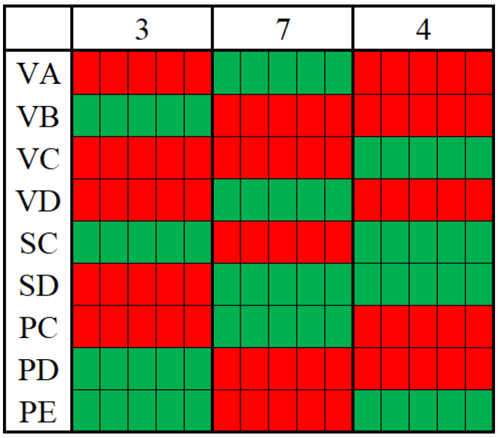
5.4 Minimální signální plán

Nejprve je potřeba si zmínit některé předpoklady - okrajové podmínky [2] týkající se délek signálů semaforů.

Minimální hodnoty signálních dob:

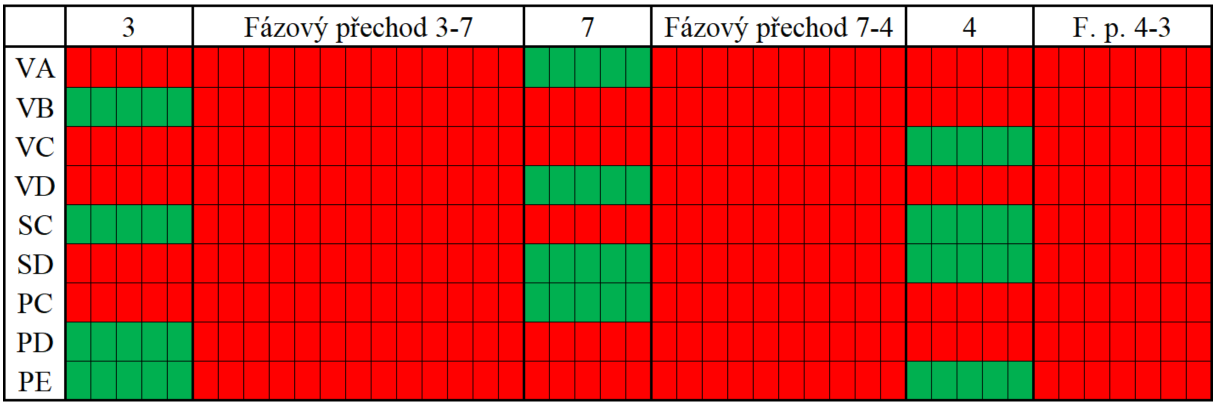
- Signál Pozor se současně svítícím červeným a žlutým světlem - 2 s,
- Signál Pozor se žlutým světlem - 3 s,
- Signál Volno - 5 s.

Nyní je možné začít sestavovat signální plány. Začne se s prvním řešením 3-7-4.
Nejprve se vytvoří popořadě kliky 3, 7, 4 o délce 5 sekund, viz obrázek 5.4. Každé políčko zde představuje 1 sekundu.



Obrázek 5.4: Kliky 3, 7, 4

Dále se mezi ně vloží, zatím pouze červené, fázové přechody o délce *příslušný mezičas + 3* sekund, viz obrázek 5.5. Příslušný mezičas znamená mezičas mezi vyklizující a najíždějící fází, viz tabulku 4.1. 3 sekundy navíc kvůli tomu, že každý signál Volno pro vozidla musí končit signálem Pozor se žlutým světlem o minimální délce 3 sekundy. Během tohoto signálu mohou vozidla stále vjet do křižovatky, takže kvůli bezpečnosti se (při momentálním sestavování signálního plánu) signál Pozor uvažuje jako signál Volno. Kdyby nějaká vyklizující fáze neobsahovala žádný signál pro vozidla, fázový přechod bude mít délku rovnou příslušnému mezičasu.

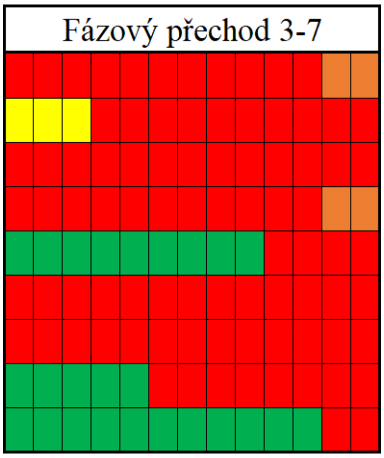


Obrázek 5.5: Vložení fázových přechodů

Nyní se vyplní fázové přechody. Začne se např. fázovým přechodem 3-7. Vezme se první proud VB vyklizující fáze 3:

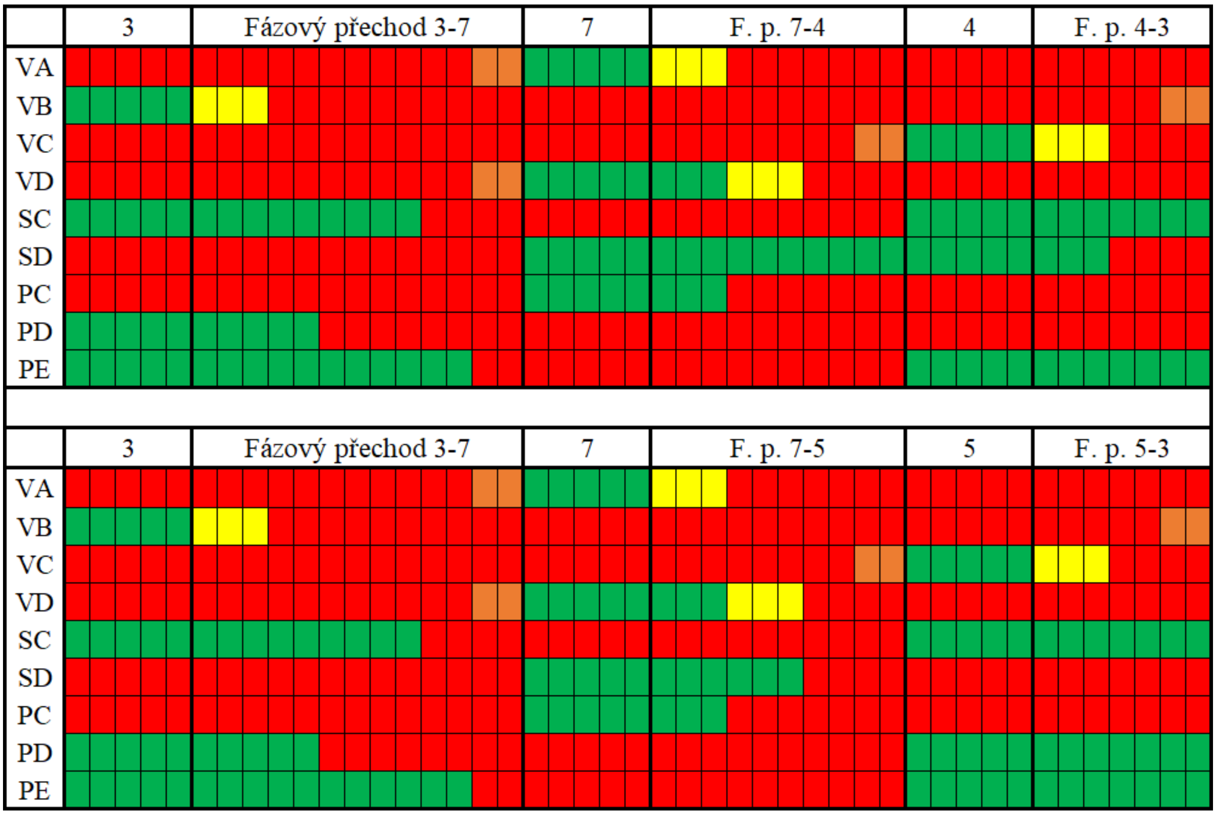
1. V tabulce mezičasů 5.6 se najde největší mezičas mezi VB a všemi proudy z následující fáze, tedy VA, VD, SD a PC. Největší mezičas je roven 10.
2. Proto se ve fázovém přechodu v řádku VB nechá 10 políček vpravo červeně.
3. Jestliže se jedná o proud pro vozidla, první 3 políčka nalevo od výše zmíněných 10 políček se vyplní žlutou. Pokud zbyla další políčka, vyplní se zelenou. Jestli se nejedná o proud pro vozidla, zbylá políčka se obarví zelenou.

Tento postup se zopakuje pro všechny zbývající proudy vyklizující kliky 3 a poté se u všech proudů pro vozidla z najíždějící kliky 7 vybarví oranžovou barvou (znázorňující signál Pozor se současně svítícím červeným a žlutým světlem) 2 políčka před začátkem této kliky, viz obrázek 5.6.



Obrázek 5.6: Fázový přechod 3-7

Analogicky se provede postup s dalšími fázovými přechody a získá se tak výsledný minimální signální plán 3-7-4. Minimální signální plán 3-7-5 se vytvoří obdobně. Obecně vytváření fázových přechodů může při vytváření minimálního signálního plánu zkrátit některé fáze z původních 5 sekund na méně sekund, neboť mají-li všechny proudy z vyklizující fáze signál Volno i ve fázovém přechodu, minimální délka 5 sekund signálu Volno je splněna i po určitém zkrácení dané vyklizující fáze. V tomto případě se tak ale nestalo. Oba zmíněné signální plány jsou vyobrazeny na obrázku 5.7.



Obrázek 5.7: Výsledné minimální signální plány

6 Diskuze výsledků

Hledání všech klik

Jelikož algoritmus prohledává všechny možnosti, je spolehlivý. Pokud má člověk k dispozici počítač s dostatečnou výbavou, jedná se o rychlé prohledání. Ruční provedení algoritmu je taky snadné, ale oproti automatizované verzi mnohokrát časově náročnější.

6.1 Generování fází

Sekvenční barvení

Algoritmus většinou nenajde optimální řešení, jelikož vygenerovaná počáteční posloupnost je úplně libovolná. Ani v tomto případě algoritmus nenašel optimální řešení, co se nejmenšího součtu mezičasů týče. Objevil ale řešení mající nejmenší možný počet fází, o což se algoritmus snaží, řešení se tedy může prohlásit za úspěšné.
Kdyby se počítalo se značně komplexnější křižovatkou, což znamená, že graf koliznosti bude větší a bude mít více hran, provést algoritmus by trvalo déle a úspěšnost algoritmu by byla pravděpodobně nižší.

Sekvenční barvení upravené

Tento algoritmus nenašel optimální řešení z pohledu mezičasů ani nejmenšího počtu fází, jelikož vygeneroval 4 fáze, zatímco nejmenší počet jsou 3 fáze. Výsledný součet mezičasů je ale u tohoto řešení srovnatelný s ostatními verzemi se 3 fázemi. Z množiny 4fázových řešení se bezpochyby jedná o jedno z nejlepších řešení.

Paralelní barvení

Algoritmus našel totožné řešení jako algoritmus sekvenčního barvení. To znamená, že také objevil optimální řešení z pohledu nejmenšího počtu fází. Avšak úspěšnost a vhodnost algoritmu je asi vyšší než u sekvenčního, což by se projevilo až při větších křižovatkách.
Na rozdíl od algoritmu sekvenčního barvení se v tomto algoritmu řadí vrcholy podle svých stupňů. Samotný algoritmus ale neošetřuje, jak vrcholy seřadit, je-li několik vrcholů se stejným stupněm. V tomto ohledu by se mohly provést nějaké modifikace a algoritmus optimalizovat, třeba seřadit zmíněné vrcholy se stejným stupněm podle intenzit.

Barvení LDF

Barvení LDF našlo jiné řešení než sekvenční, nebo paralelní. Součet mezičasů má však stejný a také našel řešení se 3 fázemi. Při větších křižovatkách je logické, že by algoritmus LDF měl být lepší než sekvenční algoritmus a srovnatelný, ne-li lepší než paralelní algoritmus kvůli určení pořadí vrcholů stupněm a barevným stupněm vrcholů.
Ačkoli má algoritmus LDF pořadí vrcholů určeno pomocí zmíněných pravidel, stále může nastat situace, kdy bude několik vrcholů mít stejný stupeň i barevný stupeň. Podobně jako u algoritmu paralelního barvení, i tu je prostor na zefektivnění.

Matematický model

Matematický model nalezl zcela jiné řešení než předchozí algoritmy. Navíc každý proud rovnou zařadil do co nejvíce fází. Výsledné součty mezičasů jsou ale rovnocenné. Záporem stále zůstává, že je potřeba předem znát minimální počet barev, takže se bez použití jiného algoritmu neobejde. Tento fakt se na druhou stranu může vzít jako pozitivum, poněvadž pak model může najít optimální řešení pro jakýkoliv počet fází, který se předem zvolí.

Model pokrývacího problému

Model našel totožné řešení jako barvící algoritmus LDF. Z pohledu minimálního počtu fází našel optimální řešení, neboť tento faktor minimalizuje. Jako vstup potřeboval znát všechny kliky, což je menší nevýhodou.

Heuristický algoritmus

Heuristický algoritmus je jako jediný schopen najít a našel více řešení. Zároveň ale nenašel všechna možná řešení, protože algoritmy sekvenčního a paralelního barvení našly jiná řešení než tento algoritmus. To znamená, že možná ani nenašel celkově optimální řešení z pohledu součtu mezičasů. Avšak pouze heuristický algoritmus objevil optimální řešení ze všech nalezených řešení. Ostatní totiž minimalizovaly počet fází, nebo maximalizovaly počet proudů ve fázích, což splnily.
Algoritmus byl proveden ručně a byl oproti ostatním algoritmům neskutečně časově náročný.

6.2 Seřazení fází

Heuristický algoritmus

Algoritmus prohledal všechny možné kombinace a pokaždé našel optimální řešení. S tím se ale zároveň pojí to, že trvá docela dlouho, než se dojde k výsledku, jelikož se provádí ručně, a tím pádem je použitelný pouze pro křižovatky s nízkým počtem fází.

Model úlohy obchodního cestujícího

Tento model najde pokaždé pouze jedno optimální řešení, i pokud existuje více optimálních řešení. Toto jedno optimální řešení je však postačující.
Mezi výhody řešení úlohy pomocí tohoto modelu oproti heuristickému algoritmu patří rychlost vyřešení. Na druhou stranu chvíli trvá, než se model nachystá, navíc je potřeba mít k dispozici počítač a optimalizační program.

6.3 Srovnání

Algoritmus sekvenčního barvení má ze všech barvících algoritmů nejméně podmínek, pomocí kterých vytváří posloupnost. Dá se tedy říci, že je pouhá náhoda, že našel řešení s nejmenším počtem barev. Matematický model umí jako jediný najít řešení pro předem daný počet fází, řešení ale nemusí být optimální. 4fázová řešení jsou vždy z pohledu nejmenšího součtu mezičasů všechna stejná, nebo horší než 3fázová, i když ne o moc.

Z výsledků lze tedy usoudit, že optimálním počtem fází je nejmenší možný počet fází. Dále většina algoritmů nenajde více řešení, což snižuje šanci najít optimální řešení.
Kdyby byly pro všechny algoritmy k dispozici naprogramované verze, bylo by mnohem jednoduší a časově efektivnější algoritmy aplikovat.

6.4 Signální plán

Signální plány, které byly vytvořeny v této práci, jsou pouze časově minimální. V případě větší efektivity je pak možné fáze časově natahovat. Ve fázových přechodech lze měnit pouze délku signálů Volno - mohou se zmenšit, nyní jsou totiž v maximální možné variantě.

7 Závěr

V bakalářské práci byly uvedeny teoretické základy z oblasti světelně řízených křižovatek a formulovala se zde úloha, která se dále pomocí získaných teoretických poznatků řešila.

Během prvních dvou kapitol byla popsána motivace celé práce, tj. proč se snaží optimalizovat fázové schéma signálního plánu křižovatky, uvedly se zde zcela základní pojmy z oblasti dopravy a komplexněji se charakterizovala úloha, jež se v práci řeší.

Třetí a čtvrtá kapitola se věnovala úlohám, na které lze části problému (generování fází a seřazení fází) převést. Mezi ně patří úloha barvení grafu, úloha pokrytí grafu nejmenším počtem klik, pokrývací problém a úloha obchodního cestujícího. Dále zde byly popsány možné postupy, algoritmy a modely řešící tyto úlohy a další problémy v celém modelu práce.

Pátá a šestá kapitola charakterizovala konkrétní model křižovatky, na který se následně aplikovaly všechny dříve popsané metody a algoritmy. Vzaly se všechny dopravní proudy křižovatky a vygenerovaly se fáze tak, aby byl počet fází co nejefektivnější. Potom se tyto výsledky vzaly, tj. podoba a počet fází, a hledalo se optimální pořadí těchto fází tak, aby se čekací doba aut co nejvíce minimalizovala. Byla objevena 2 optimální řešení. V konečné fázi se z těchto řešení vytvořily 2 optimální minimální signální plány. Poté se veškerá objevená řešení a použité algoritmy mezi sebou porovnaly a vyhodnotily se.

V této práci se dospělo k několika závěrům. Barvící algoritmy nejsou moc vhodné k řešení definovaného modelu, rozhodně mají prostor ke svému zefektivnění. Heuristický algoritmus pro generování fází najde vždy optimální řešení a zároveň i více řešení. Všechny ručně provedené algoritmy jsou náročné na provedení, v některých případech se vyplatí použít algoritmy, na které je potřeba výpočetní technika. Optimálním počtem fází je minimální možný počet fází.

Algoritmy na generování fází použité v této práci nezohledňují minimalizaci mezičasů. Hledá se pouze optimální řešení z pohledu příslušných úloh - minimální počet barev, každý vrchol obarven maximální počtem barev a zároveň alespoň jednou barvou. Tento krok může vygenerovat různé kombinace fází a odlišné kombinace fází pak mají jiné mezičasy. Jinak řečeno, výsledné mezičasy závisí už na tom, jak se fáze vygenerují. Pro další možnosti této práce je tedy v tomto ohledu možné metody zefektivnit, tj. zohlednit minimalizaci mezičasů už v počáteční fázi řešení.

Celá práce uvažovala jako primární kritérium minimalizaci mezičasů. Existují ale i další kritéria jako jsou například čekací čas, délka fronty, míra využití pruhu, emise, počet zastavení atd., která byla zmíněna v předpokladech modelu. Tato kritéria se zohlednila pouze minimálně, tím pádem se z jejich pohledu pravděpodobně vynechala optimální řešení. V další práci by pak bylo možné problém definovat s více kritérii.

Literatura

[1] HOVORKA, Radek. *První elektrický semafor řídil provoz už před sto lety.* [online]. 2014 [cit. 2018-03-16]. Dostupné z: `https://auto.idnes.cz/prvni-semafor-0dw-/automoto.aspx?c=A140617_225148_automoto_fdv`

[2] *Navrhování světelných signalizačních zařízení pro řízení provozu na pozemních komunikacích: technické podmínky: TP 81* [online]. 3. vyd. Praha: Ministerstvo dopravy, 2015. [cit. 2018-03-19].
Dostupné z: `http://www.pjpk.cz/data/USR_001_2_8_TP/TP_81.pdf`

[3] PALÚCH, Stanislav. *Algoritmická teória grafov* [online]. Žilina: Žilinská univerzita, 2008 [cit. 2018-03-19].
Dostupné z: `http://frcatel.fri.uniza.sk/users/paluch/grafy.pdf`

[4] RUSEK, Michal. Aplikace metod barvení grafů pro určení minimálního počtu fází světelně řízených křižovatek. *Perner's Contacts* [online]. 2011, 6(1), 275-285. [cit. 2018-03-19]. ISSN 1801-674X.
Dostupné z: `http://pernerscontacts.upce.cz/21_2011/Rusek.pdf`

[5] ČERNÁ, Anna a Jan ČERNÝ. *Teorie řízení a rozhodování v dopravních systémech.* Pardubice: Institut Jana Pernera, 2004. ISBN 80-86530-15-9.

[6] RUSEK, Michal. Aplikace úlohy obchodního cestujícího pro výběr optimálního pořadí fází světelně řízených křižovatek. *Perner's Contacts* [online]. 2011, 6(4), 326-334. [cit. 2018-03-19]. ISSN 1801-674X.
Dostupné z: `http://pernerscontacts.upce.cz/23_2011/Rusek.pdf`

[7] *Ostravské komunikace, a.s.* [online]. [cit. 2018-04-17]. Dostupné z: `http://okas.cz/userfiles/reference_pdf/LISA.pdf`

[8] WILDMANN, Jeffrey. maximalCliques. *Drexel University* [online]. 2011 [cit. 2018-04-18]. Program je snadno dostupný z: `https://www.mathworks.com/matlabcentral/fileexchange/30413-bron-kerbosch-maximal-clique-finding-algorithm`

[9] BURKARDT, John. tsp_brute. *Florida State University, Dept. of Scientific Computing* [online]. 2013 [cit. 2018-04-23]. Dostupné z: `https://people.sc.fsu.edu/~jburkardt/m_src/tsp_brute/tsp_brute.html`

Přílohy

CD s textovým souborem a programy v GAMS a Matlab.

- Matematicky_model.gms
- Set_Cover_Problem.gms
- maxCliques.m [8]
- tsp_brute.m [9]
- tsp_vstup.txt